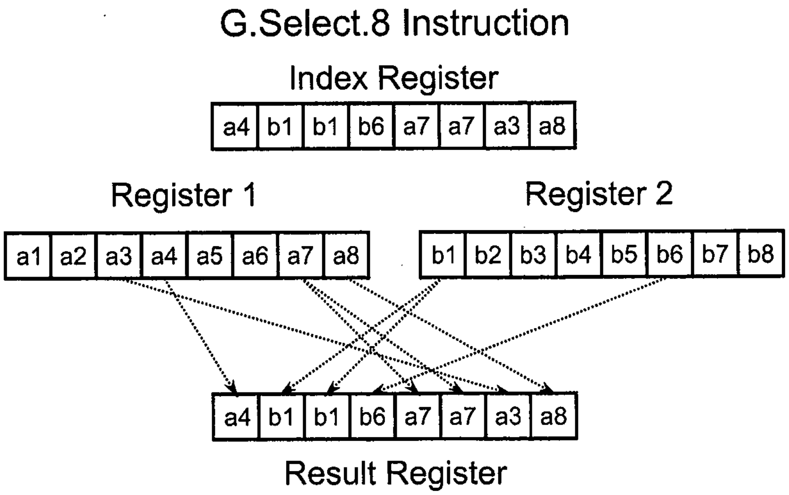
REMARKS

Claims 1-12, 14-25, 40, 41, 43, 45-47, 50, 52, 54-57, 59-61, 64, and 66 are currently pending in this application. By this response, independent claims 1, 14, 40, 50, 54, and 64 are amended, and claims 13, 26, 42, 44, 48, 49, 51, 53, 58, 62, 63, 65, and 67 are canceled without prejudice. Support for the amendments is found in the specification, including the claims, as originally filed. No new matter has been introduced. Favorable reconsideration of the application in light of the foregoing amendments and following comments is respectfully solicited.

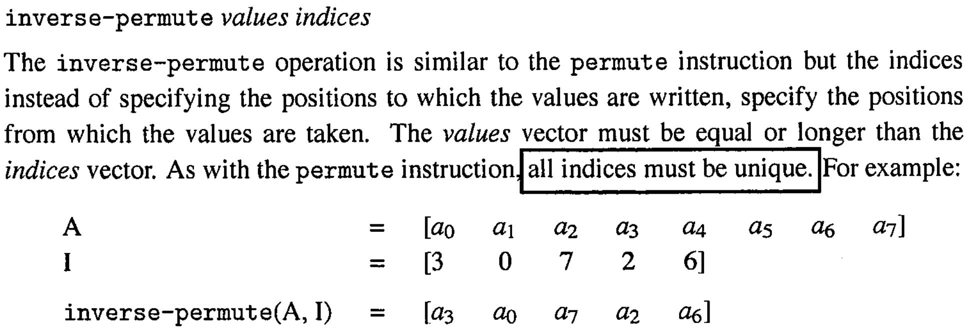
I. INTERPRETATION OF THE CLAIMS

Section 28 of the Office Action adopted an interpretation of the claims "requiring that each selection is performed independently based on only 1 index" (*see* Office Action, page 15, lines 3-6). Although Applicants disagree that this was a reasonable interpretation of the claims as previously presented, in the interest of expediting prosecution of this application, Applicants have amended the independent claims for the purpose of clarification. For example, amended independent claim 1 recites "the data selection operand comprising a plurality of fields each ~~independently~~ selecting any one of the plurality of data elements and each field having a value not restricted by the other fields included in the data selection operand."

Much as noted previously by Applicants, the G.Select.8 instruction disclosed by this application provides an example of the claimed subject matter:


G.Select.8 Instruction

As can be seen in the above illustrative figure, fields in the index register specified by the G.Select.8 instruction are not required to have unique values. For example, the indices for elements a7 and b1 are each used twice in the index register shown above. In contrast, as discussed below, the inverse-permute operation disclosed by Blelloch, page 66, expressly requires that "all indices must be unique."


`inverse-permute` *values indices*

The `inverse-permute` operation is similar to the `permute` instruction but the indices instead of specifying the positions to which the values are written, specify the positions from which the values are taken. The *values* vector must be equal or longer than the *indices* vector. As with the `permute` instruction, all indices must be unique. For example:

$$\begin{array}{lcllllllll}
\text{A} & = & [a_0 & a_1 & a_2 & a_3 & a_4 & a_5 & a_6 & a_7] \\
\text{I} & = & [3 & 0 & 7 & 2 & 6] & & & \\
\texttt{inverse-permute(A, I)} & = & [a_3 & a_0 & a_7 & a_2 & a_6] & & &
\end{array}$$

The reason for this difference between the claimed subject matter and Blelloch is due to a fundamental difference in the architectures used by Blelloch to implement the instructions it describes and that addressed by the claimed subject matter. Blelloch's implementations are expressly described as multiprocessor systems. On such systems, issues arise from inter-processor communications which prevent using nonunique indices without introducing delay, congestion, or ensuring lack of parallelism. Seeking to avoid such issues, Blelloch expressly indicates that "all indices must be unique."

Section 1.3 of Blelloch, beginning on page 9, entitled "Implementation" explains that it

> . . . outlines how a V-RAM can be implemented on two architectures: a shared memory SIMD multiprocessor and a distributed memory MIMD multiprocessor. As examples, it considers two vector instructions: an elementwise add (`p+`) and a `permute`. . . . Details of how to implement a V-RAM on the Connection Machine, a distributed memory SIMD multiprocessor, are discussed in Chapter 12.
>
> (*emphasis added*)

Looking at the "shared memory SIMD multiprocessor" implementation, page 9 details:

> . . . assigning each processor to an independent block of each vector. When executing a vector instruction on a vector a, each processor loops over the elements of a it is responsible for. . . . For a permute instruction, each processor reads a value and index for its elements, and writes the value in the destination vector at the position specified by the index.

From this description, it can be seen why Blelloch, page 66 requires that for the inverse-permute instruction "all indices must be unique," particularly when "providing in parallel the data elements selected by the fields," as recited in claim 1, for example. As one processor is assigned to a given element, if that element were repeated it could not be provided by the one processor to 2 or more positions in parallel.

Looking at the "distributed memory MIMD multiprocessor" implementation, page 9 details:

> Each vector is evenly divided among the processor memories. The processors run in single program multiple data (SPMD) mode and each processor is responsible for the elements in its own memory. . . . The permute instruction requires sending data to other processors—based on the index each source processor determines the destination processor for each of its elements and sends the elements to those processor.

From this description, it can be seen why Blelloch, page 66 requires that for the inverse-permute instruction "all indices must be unique," particularly when "providing in parallel the data elements selected by the fields," as recited in claim 1, for example. As one source processor is assigned to a given element, and the elements of the destination have corresponding destination processors, if that element were repeated it could not be provided by the one source processor to 2 or more destination processors in parallel.

Chapter 12 describes implementation on the CM-2 Connection Machine, which Blelloch, page 190 describes as "a fine-grained data-parallel computer with between 8K and 64K processors." In other words, it is also a multiprocessor system. Blelloch, pages 192-93 describe implementation of permutation instructions in the described V-RAM implementation. From this description, it can be seen why Blelloch, page 66 requires that for the inverse-permute instruction "all indices must be unique," particularly when "providing in parallel the data elements selected by the fields," as recited in claim 1, for example. Page 193 expressly indicates that a problem in inter-processor communication occurs by "having many elements in one slice going to a single processor and congesting the routing hardware."

Thus, to avoid inherent limitations of multiprocessor systems used by Blelloch, Blelloch expressly instructs that "all indices must be unique" for its inverse-permute operation, making the disclosed inverse-permute operation very inefficient relative to the claimed subject matter, which overcomes and does not require such a limitation.

A. The Office Action improperly read the term "independently" out of the claims

In formulating the anticipation rejection based on Blelloch, the Office Action appears to read the term "independently" out of the claims. When reciting the limitations of claim 1 and pointing to allegedly corresponding features in Blelloch, the Office Action simply left out the term "independently," as if it had not been presented in the claim (*See* Office Action, at pp. 3-4, paragraph 6):

> Referring to claim 1, Blelloch discloses, as claimed . . . decoding a single instruction (Inverse-permute; see page 66) for selectively arranging data,

> specifying a data selection operand (see Vector File address format in Fig. 13) . . ., the data selection operand comprising a plurality of fields (Each element of Index Vector I; see page 66) each selecting one (such as 3, 0, 7, 2 and 6 as shown in the last figure of page 66) of the plurality of data elements . . .
>
> (*emphasis added*)

However, anticipation under Section 120 requires that a single reference discloses, expressly or inherently, each and every limitation of a claim. Applicants respectfully request that all limitations of the claims be considered when assessing the patentability of the claimed subject matter, including, for example, "each field having a value not restricted by the other fields included in the data selection operand," as recited in independent claim 1.

B. The Office Action improperly interpreted the claims as having a "unique indices" requirement

Furthermore, the Office Action improperly interpreted the claims as having a "unique indices" requirement. The Office Action states at page 15, paragraph 28: "Applicant has classified the G.SELECT.8 as a 'Group Permute' function (*See* Exhibit 1, page 80, filed January 8, 2009). Permutation is merely reordering and would therefore require that all of the indexes (sic) be unique (in order to be a 'permute' function)." This strained interpretation contradicts the actual description of the operation of the G.Select.8 instruction. Instead, it relies solely upon a label that was once used for the G.Select.8 instruction, which is not determinative. Furthermore, such an interpretation disregards the plain language of the claims. These reasons are explained in detail below.

1. A requirement "that all of the ind[ices] be unique" would directly contradict the "plain meaning" of the claims

As noted in MPEP § 2111.01(I), "the words of the claim must be given their plain meaning unless the plain meaning is inconsistent with the specification." In this application, claim 1, for example, recites "each field having a value not restricted by the other fields included in the data selection operand." In contrast, to "require that all of the ind[ices] be unique," as proposed at page 15, lines 9-10 of the Office Action, would mean that the selection of a value for each field must necessarily take into account, or be restricted by, the values selected for the other fields, in order to ensure the field values are unique (*i.e.*, "that all of the ind[ices] are unique"). This is directly contrary to the plain meaning of the claimed limitation, would not be a correct interpretation of the claims, and cannot be adopted.

2. A requirement "that all of the ind[ices] be unique" would be inconsistent with the broad disclosure in the specification

Additionally, an interpretation of the claims requiring unique indices would contradict the disclosure of the claimed subject matter in the specification. As noted previously, the G.Select.8 instruction, which is disclosed in the specification, provides an example of the claimed subject matter. Applicants have carefully documented the operation of the G.Select.8 instruction. In particular, the Euterpe MicroArchitecture Manual describes the operation of this instruction in full detail, down to the bit level.[1] The Euterpe MicroArchitecture Manual clearly shows that the G.Select.8 instruction takes a data selection operand having multiple fields, with each field having a value not restricted by the other fields included in the data selection operand. Sections 5-8 of the Second Declaration of Craig Hansen (an inventor of the present application) Under 37 CFR § 1.131 (Hansen II declaration), filed on September 21, 2009, explains the Euterpe MicroArchitecture Manual's description of G.Select.8 instruction, in part, as follows:

> . . . In this fashion, the location of each 8-bit byte in the destination register is determined by a 4-bit control field, with control bits 0-3 selecting an 8-bit byte from any location in the source register, control bits 4-7 selecting an 8-bit byte from any location in the source register, including the location previously

[1] Various versions of the Euterpe MicroArchitecture Manual have been submitted to the Patent Office. One version dated prior to the filing date of the present application was submitted on September 21, 2009 as Exhibit A for the Second Declaration of inventor Craig Hansen under Rule 1.131. A later version of the manual was submitted as part of the original disclosure of the present application. No significant changes were made to the description of the G.Select.8 instruction across these various versions of the manual.

> selected, and placing it in bits 8-15 of the destination register, and so on until 128 bits consisting of 16 independently selected bytes have been selected from the source register and placed in the destination register as determined by the 4-bit control bits corresponding to each destination byte location . . ."
>
> Hansen II declaration, page 4 (*emphasis added*)

The claim interpretation proposed by the Office Action, requiring such fields to represent unique indices, contradicts the actual description of the G.Select.8 instruction. In this context, a unique indices requirement would force the condition that control bits 0-3, control bits 4-7, and so on, must take on different values. For example, if control bits 0-3 took on a value of "0001," control bits 4-7 would not be allowed to take on a value of "0001." However, the description of the G.Select.8 instruction clearly does not impose such a restriction. Thus, the interpretation proposed by the Office Action would directly contradict the actual description of the G.Select.8 instruction provided in the Euterpe MicroArchitecture Manual and included in the specification of this application.

Indeed, the absence of any mention of a supposed unique indices requirement in the G.Select.8 documentation is quite telling. Assuming, for the sake of discussion, the G.Select.8 instruction truly required unique indices (which it does not), one would expect to find ample documentation of such a requirement.[2] Absent such documentation, a user would have no idea such a specific requirement exists and might use the instruction improperly, *i.e.*, by supplying the instruction with an operand specifying indices that are not unique. However, nowhere in the Euterpe MicroArchitecture Manual's description of the G.Select.8 instruction is there any mention of a requirement for unique indices. Clearly, the G.Select.8 instruction simply has no requirement for unique indices. It is unreasonable to reach the opposite conclusion – that a

[2] Shortly after the Introduction portion, the Euterpe MicroArchitecture Manual states that "[n]othing in this specification should be construed to limit the implementation choices of the conformant system beyond the specific requirements stated herein."

detailed micro architecture manual describing the exact operation of an instruction would keep such a crucial attribute of the instruction as an undocumented secret.

3. Interpretation of the claims in view of the Cray Research presentation is improper

Page 15, lines 7-9 of the Office Action relies upon page MU0020396 of Exhibit 1 (the "Cray Research presentation") of the Declaration of Craig Hansen Under 37 C.F.R. § 1.131 (Hansen I declaration) filed on January 8, 2009 as a basis for concluding that the claims as previously presented "require that all of the ind[ices] be unique." However, the Cray Research presentation forms no part of the disclosure for this application, but instead was submitted as an example of conception of the claimed subject matter prior to the effective date of the Lee reference. The Cray Research presentation does not serve as intrinsic or extrinsic evidence upon which an interpretation may be constructed – particularly given that limiting the claim to a permute operation (*i.e.*, in which all indices must be unique) would be inconsistent with the language of the claims and the specification.

4. Even if the Cray Research presentation were relevant to interpretation of the claims, its broad description of the instruction contradicts the Office Action's interpretation

Page 5, lines 7-10 of the Office Action relies upon the Cray Research presentation presenting the G.Select.8 instruction on one of three sheets labeled "Group Permute." However, this mere labeling is not determinative. Eschewing the actual description of the operation of the G.Select.8 instruction, the Office Action instead focuses its attention on a label that Applicants once used to refer to a collection of instructions that included the G.Select.8 instruction. Specifically, the Office Action points to the label "Group Permute" found in the Cray Research presentation *See* Office Action, p. 15, paragraph 28. Whatever meaning the Office Action ascribes to such an indirect label once used for the G.Select.8 instruction, it does not outweigh the actual disclosure of the instruction in the Cray Research presentation. Sections 3-4 of the Hansen II declaration explains why "[o]ne skilled in the art would recognize . . . that the rearrangement is done with a plurality of fields each independently selecting one of the plurality of data elements." This explanation also demonstrates how the Cray Research presentation discloses "each field having a value not restricted by the other fields included in the data selection operand," as recited in claim 1, for example.

Moreover, Applicants have long since abandoned use of the term "Group Permute" as a label associated the G.Select.8 instruction. For example, in the draft version of the Euterpe MicroArchitecture Manual dated prior to August 1, 1995, Applicants clearly avoided use of the term "Group Permute" to label the G.Select.8 instruction in any way.[3] Similarly, in the entire disclosure of the present application, Applicants have avoided any use of the term "Group Permute" to label the G.Select.8 instruction. Thus, the history of Applicants' usage of the term "Group Permute" actually indicates that, starting from a time prior to the filing of the present application, and even prior to the August 1, 1995 date for Lee, Applicants have consistently turned away from associating the term "Group Permute" with the G.Select.8 instruction. If anything, this history shows that whatever connotation the term "Group Permute" might have, it was long ago rejected by Applicants, and should not be applied to the G.Select.8 instruction.

C. The application properly supports the claimed "each field having a value not restricted by the other fields included in the data selection operand," to satisfy requirements under 35 U.S.C. § 112

The present disclosure properly supports the claimed "each field having a value not restricted by the other fields included in the data selection operand," to satisfy the requirements of 35 U.S.C. § 112, first paragraph. The Office Action warned that Applicants' interpretation of

[3] *See* the draft version of the Euterpe MicroArchitecture Manual (at p. 23), submitted as "Exhibit A" to the Rule 1.131 Declaration of inventor Craig Hansen dated 9/17/09, in response to the 102(e) rejection based on Lee.

the claims as previously presented would lead to a lack-of-support issue under 35 U.S.C. § 112, first paragraph. *See* Office Action at p. 15, paragraph 28 ("If the claim were to be interpreted in this manner, the specification would not provide support under 35 U.S.C. § 112, first paragraph"). Applicants respectfully disagree.

Applicants' claim interpretation merely reflects the plain language of the claims, *i.e.*, that each field has a value not restricted by other fields. An embodiment of such a claimed feature is the G.Select.8 instruction, which is described in detail, down to the bit level, in the Euterpe MicroArchitecture Manual.[4] Such detailed descriptions provide sufficient support for the claimed "each field having a value not restricted by the other fields included in the data selection operand," and satisfy the requirements of Section 112.

In any event, if the Patent Office's position is that the claimed invention lacks support in the disclosure, a separate rejection under 35 U.S.C. § 112, first paragraph, should have been issued. Such a rejection should be not be entangled with the current Section 102(b) anticipation rejection based on Blelloch. As it stands, it is difficult for Applicants to fully respond to a rejection under 35 U.S.C. § 112, first paragraph, that is hinted at by the Office Action, but not fully set forth.

II. REJECTIONS UNDER 35 U.S.C. §§ 102 AND 103 IN VIEW OF BLELLOCH

In section 5 of the Office Action, claims 1, 2, 4-8, 11-15, 17-21, 24-26, 40, 41, 43, 46, 50, 54, 55, 57, 60, and 64 were rejected under 35 U.S.C. § 102(b) as being anticipated by Blelloch (*Vector Models for Data-Parallel Computing*). Claims 3, 9, 10, 16, 22, 23, 42, 44, 45, 47-49, 51-53, 56, 58, 59, 61-63, and 65-67 were rejected under 35 U.S.C. § 103(a) as being unpatentable

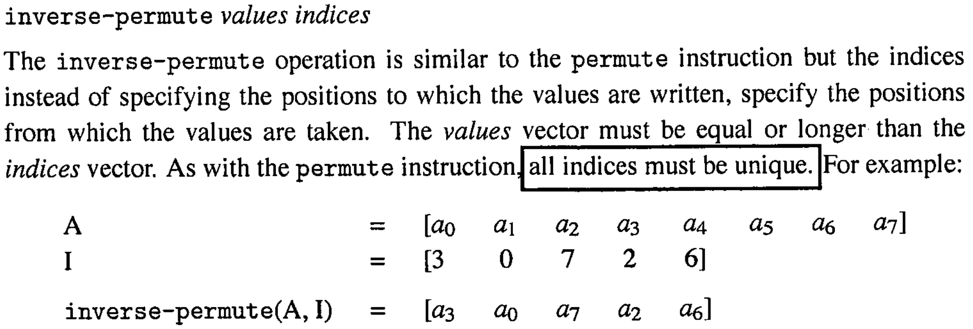
[4] A copy of the Euterpe MicroArchitecture Manual was filed as part of the disclosure of the present application.

over Blelloch in view of *In re Rose*, 105 USPQ 237 (CCPA 1955). Applicants respectfully traverse.

An anticipation rejection under 35 U.S.C. § 102(b) must illustrate that every limitation found in the claim is disclosed by the cited reference. There are a number of claimed limitations which Blelloch does not disclose. For at least this reason, the current anticipation rejection under Section 102(b) based on Blelloch must be withdrawn.

A. Blelloch does not disclose a data selection operation utilizing a data selection operand comprising a plurality of fields with "each field having a value not restricted by the other fields included in the data selection operand"

Claim 1 recites, *inter alia*, "the data selection operand comprising a plurality of fields each selecting any one of the plurality of data elements and each field having a value not restricted by the other fields included in the data selection operand." In contrast, Blelloch expressly requires that the indices of the index vector for the inverse-permute operation "must be unique," as indicated on page 66:



`inverse-permute` *values indices*

The `inverse-permute` operation is similar to the `permute` instruction but the indices instead of specifying the positions to which the values are written, specify the positions from which the values are taken. The *values* vector must be equal or longer than the *indices* vector. As with the `permute` instruction, all indices must be unique. For example:

$$\begin{array}{lcllllllll} A & = & [a_0 & a_1 & a_2 & a_3 & a_4 & a_5 & a_6 & a_7] \\ I & = & [3 & 0 & 7 & 2 & 6] & & & \\ \texttt{inverse-permute}(A, I) & = & [a_3 & a_0 & a_7 & a_2 & a_6] & & & \end{array}$$

Blelloch clearly discloses that the value of an index is restricted by the other indices. This is a fact that the Office Action cannot, and does not, refute. According to Blelloch, the value of each index must be selected by taking into account the values of the other indices, as to not repeat any value selected by another index. Thus, each index in Blelloch is restricted by values selected for all the other indices. As discussed on pages 17-19 above, Blelloch expressly requires that "all indices must be unique" due to fundamental differences between the architectures used by Blelloch and that addressed by the claimed subject matter. Clearly, the teachings of Blelloch do not square with the presently claimed subject matter, which requires "each field having a value not restricted by the other fields." Thus, it is clear that Blelloch does not disclose, or even render obvious, the claimed subject matter.

B. Blelloch fails to disclose "the instruction independently specifying the first register and the second register"

Additionally, claim 1 recites, *inter alia*, "the instruction independently specifying the first register and the second register, the first and second registers providing a plurality of data elements." An example of these limitations is illustrated by the G.Select.8 instruction described in the Euterpe MicroArchitecture Manual included in the disclosure for the present application. The disclosed G.Select.8 instruction includes two independently specified source registers, r_a and r_b, which together provide sixteen 8-bit data elements which may be selected by the fields included in destination register r_c. Paragraph 13 of the Third Declaration of Craig Hansen Under 35 U.S.C. § 1.131 (the Hansen III declaration) submitted herewith further describes the G.Select.8 instruction, with multiple independently specified source registers.

By providing a single instruction in which two source registers are independently specified, the claimed subject matter gives flexibility and provides for efficiencies that Blelloch's inverse-permute operation does not have. The flexibility is derived from the ability to select elements from a plurality of registers rather than a single vector, and efficiencies derive from the ability to perform a series of G.Select.8 operations and by changing only one of the specified registers, change the contents of the table for the next operation. Overall processing speed can be enhanced by varying only one of the registers, rather than creating an entirely new source vector, as in Blelloch.

Section 6 of the Office Action concluded that "First and Second halves of A" (although Blelloch does not denote the "halves" suggested by the Office Action) disclosed the "first and second register," as previously presented. However, the inverse-permute operation shown on Blelloch, page 66 only specifies a single values vector, and does not provide an "instruction independently specifying the first register and the second register," as recited in claim 1, as the "halves" suggested by the Office Action are not independently specified. Thus, Blelloch does not disclose or suggest the claimed subject matter.

C. Conclusion

For at least the above reasons, Applicants respectfully submit that Blelloch fails to disclose or render obvious the subject matter recited in claim 1. For much the same reasons, independent claims 14, 40, 50, 64, and 64 are also neither disclosed nor rendered obvious by Blelloch. At least by virtue of their dependency upon the independent claims, dependent claims 2-12, 15-25, 41, 43, 45-47, 52, 54-57, 59-61, and 66 are neither anticipated nor rendered obvious by Blelloch. *In re Rose*, which is cited with respect to limitations recited in dependent claims, does not bridge the above gaps between the claims and Blelloch. Thus, Applicants respectfully request withdrawal of the rejections under Sections 102 and 103 in view of Blelloch.

III. REJECTIONS UNDER 35 U.S.C. §§ 102 AND 103 IN VIEW OF LEE

Claims 1, 11, 14, and 24 were rejected under 35 U.S.C. § 102(e) as being anticipated by Lee (U.S. Patent No. 6,381,690). Claims 12 and 25 were rejected under 35 U.S.C. § 103(a) as being unpatentable over Lee in view of Matsuura (U.S. Patent No. 4,725,973). Applicants respectfully traverse.

Claims 1, 11, 14, and 24 are rejected under 35 U.S.C. § 102(e) as being anticipated by Lee (U.S. Patent No. 6,381,690). However, Lee is not prior art to the present invention. In previous responses, Applicants submitted swear-behind declarations under 37 C.F.R. 1.131 filed on January 8, 2009 from the inventors Mr. Craig Hansen and Dr. John Moussouris, as well as the second, updated Hansen II declaration. These declarations establish that Lee is not prior art to the present invention. The only objection that the current Office Action raises regarding the swearing-behind of Lee is a concern that the available evidence supposedly does not show that data elements are provided in parallel in a catenated result. *See* Office Action, page 2, paragraph 2.

Filed concurrently with this Amendment is the Hansen III declaration, explaining evidence of Applicants' conception of providing in parallel the data elements selected by the fields to respective predetermined positions in a catenated result prior to August 1, 1995 (the effective date for Lee) illustrated by:

- Exhibit 1 of the Hansen I declaration (paragraphs 3-4)
- Exhibit A of the Hansen II declaration (paragraphs 5-6)
- Exhibit 2 of the Hansen I declaration (paragraphs 7-10)

The Hansen III declaration demonstrates Applicant's conception of the claimed subject matter, including "providing in parallel the data elements selected by the fields to respective predetermined positions in a catenated result."[5]

In view of the above, Applicants respectfully submit that the swear-behind declarations under 37 C.F.R. 1.131 from Mr. Hansen and Dr. Moussouris properly establish that Lee is not

[5] Additionally, paragraph 12 of the Hansen III declaration illustrates evidence of conception of the subject matter recited in claims 12 and 25, as section 2 of the Office Action incorrectly concluded that the previously submitted evidence was insufficient to establish conception of these claims. Also, paragraph 13 of the Hansen III declaration

available as prior art for the claimed subject matter. Thus, the rejections in view of Lee should be withdrawn.

IV. CONCLUSION

In view of the foregoing, Applicants submit that all claims now pending in this Application are in condition for allowance. The issuance of a formal Notice of Allowance at an early date is respectfully requested. If the Examiner believes a telephone conference would expedite prosecution of this application, please telephone the undersigned at telephone number indicated below.

To the extent necessary, a petition for an extension of time under 37 C.F.R. 1.136 is hereby made. Please charge any shortage in fees due in connection with the filing of this paper, including extension of time fees, to Deposit Account 500417 and please credit any excess fees to such deposit account.

Respectfully submitted,

McDERMOTT WILL & EMERY LLP

Eric M. Shelton
Registration No. 57,630

600 13th Street, N.W.
Washington, DC 20005-3096
Phone: 202.756.8000 EMS:MWE
Facsimile: 202.756.8087
Date: June 22, 2010

Please recognize our Customer No. 20277 as our correspondence address.

illustrates evidence of conception of a single instruction independently specifying the first register and the second register.